

Haase, Knut

Working Paper — Digitized Version

Capacitated lot-sizing with linked production quantities of adjacent periods

Manuskripte aus den Instituten für Betriebswirtschaftslehre der Universität Kiel, No. 334

Provided in Cooperation with:

Christian-Albrechts-University of Kiel, Institute of Business Administration

Suggested Citation: Haase, Knut (1993) : Capacitated lot-sizing with linked production quantities of adjacent periods, Manuskripte aus den Instituten für Betriebswirtschaftslehre der Universität Kiel, No. 334, Universität Kiel, Institut für Betriebswirtschaftslehre, Kiel

This Version is available at:

<https://hdl.handle.net/10419/155411>

Standard-Nutzungsbedingungen:

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

Terms of use:

Documents in EconStor may be saved and copied for your personal and scholarly purposes.

You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.

If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.

No. 334

Capacitated Lot-Sizing with Linked Production

Quantities of Adjacent Periods

Knut Haase

December 1993

Knut Haase, Institut für Betriebswirtschaftslehre, Christian-Albrechts-Universität zu Kiel,
Olshausenstr. 40, 24118 Kiel

Abstract: The well-known Capacitated Lot-Sizing Problem (CLSP) is based on the assumption that for each lot in a period setup costs are incurred. The setup costs of a CLSP solution can be reduced by linking the production quantities of an item which is scheduled in two adjacent periods. Therefore we propose the CLSP with linked lot-sizes of adjacent periods (semi-sequencing), denoted by CLSPLA.

The CLSPLA is formulated as a mixed-integer programming model. For the heuristic solution of the CLSPLA we provide a method which is **backward** oriented, denoted by BACLSPLA. We apply a priority rule which consists of a convex combination of holding and setup costs. The convex combination as well as the solution quality depend on a parameter value from $[0,1]$. We perform a simple parameter value search to obtain a low costs solution.

The CLSP may be solved by the famous Dixon-Silver heuristic. A CLSP-schedule (of the Dixon-Silver heuristic) can be transformed into a CLSPLA solution. A computational study shows, that BACLSPLA, i.e. the integration of (semi-) sequencing, is more efficient.

Keywords: Lot-sizing, semi-sequencing, Dixon-Silver heuristic, backward method

1. Introduction

We characterize the deterministic lot-sizing problem which is addressed: A single-stage system is considered, where a number of different items $j=1,\dots,J$ have to be manufactured on one machine (corresponding to a single capacity constraint). The time horizon T is segmented into a finite number of time periods $t=1,\dots,T$. The machine is available with C_t capacity units in period t . Producing one unit of item j absorbs p_j capacity units (finite production speed). The demand for item j in period t , d_{jt} , has to be satisfied without delay (shortages are disallowed). To setup the machine for item j causes setup costs s_j . Inventory costs per unit h_j (holding costs coefficient) are incurred for the inventory of item j at the end of a period. The objective is to minimize the sum of setup and holding costs.

In literature heuristics (cf. [5], [6], [8], [10], [11], [3]) and exact methods (cf. [1], [7]) are proposed. The methods are developed for the so-called capacitated lot-sizing problem (CLSP), which is based on the assumption

"Setup costs occur for each lot in a period".

We define the decision variables

I_{jt} = the inventory of item j at the end of period t ,

q_{jt} = the quantity (lot-size) of item j to be produced in period t , and

x_{jt} = a binary variable indicating whether setup occurs for item j in period t ($x_{jt}=1$) or not ($x_{jt}=0$).

Thus, mathematically the CLSP can be stated as follows:

$$\min \sum_{j=1}^J \sum_{t=1}^T (s_j x_{jt} + h_j I_{jt}) \quad (1)$$

subject to

$$I_{j,t-1} + q_{jt} - I_{jt} = d_{jt} \quad (j=1,\dots,J; t=1,\dots,T) \quad (2)$$

$$\sum_{j=1}^J p_j q_{jt} \leq C_t \quad (t=1,\dots,T) \quad (3)$$

$$C_t x_{jt} - p_j q_{jt} \geq 0 \quad (j=1,\dots,J; t=1,\dots,T) \quad (4)$$

$$I_{jt}, q_{jt} \geq 0 \quad (j=1,\dots,J; t=1,\dots,T) \quad (5)$$

$$x_{jt} \in \{0, 1\} \quad (j=1,\dots,J; t=1,\dots,T) \quad (6)$$

W.l.o.g. it is $I_{j0}=0$ for all $j=1,\dots,J$.

The objective to minimize the total costs is expressed by (1). The inventory at the end of a period is obtained by the equations (2). Constraints (3) ensure that the total production in each period does not exceed the capacity. Constraints (4) force for each lot (greater than 0) the binary setup variable to be one thus increasing the sum of setup costs. The suitable domains of

the variables are determined by the restrictions (5) and (6). The non-negativity condition of the inventory ensures that a shortage does not occur.

To motivate a modification of the CLSP we consider Example 1.

Example 1: Let $J=2$, $T=3$, $s_1=s_2=100$, and $h_1=h_2=1$. For each period t the demand d_{jt} of the item j and the capacity C_t are given in Table 1 (missing entries are zero):

Table 1				Table 2			
t	1	2	3	t	1	2	3
d_{1t}	5		6	q_{1t}	5 \cup	6	
d_{2t}		3	4	q_{2t}		3 \cup	4
C_t	10	10	10				

The optimal solution of the CLSP with objective function value $Z_{CLSP}^*=300$ is determined by equalizing lot-sizes and demands, i.e. $q_{jt}=d_{jt}$ for all $j=1,...,J$, and $t=1,...,T$. Let us now consider the solution in Table 2 where " \cup " denotes the linking of the production quantities of adjacent periods, e.g. in the second period we start with the item $j=1$. Hence, it is not necessary to change the setup state of the machine because it was already prepared for item $j=1$ at the end of the period $t=1$. The total costs are reduced to $s_1 + s_2 + 6 \cdot h_1 = 2 \cdot 100 + 6 \cdot 1 = 206$. Thus, the CLSP solution quality can be poor in the case where the setup state can be preserved between adjacent periods. Furthermore, a solution approach for lot-sizing and semi-sequencing, i.e. determining the last and the first item in a period, may provide a better solution quality than an algorithm for the CLSP.

Before presenting a lot-sizing and semi-scheduling approach, in the next section we modify the CLSP regarding the possibility to link lot-sizes of adjacent periods. Then, for the heuristic solution a simple method which is backward oriented and which relies on a priority rule will be introduced in Section 3. In Section 4 a computational study is performed where the heuristic is compared for large instances with the Dixon-Silver heuristic.

2. The CLSP with Linked Lot-Sizes of Adjacent Periods (CLSPLA)

In practical cases where the setup state of a production facility can be preserved from one period to the succeeding period the CLSP may be not well-suited due to the basic assumption of the CLSP which implies that the setup state at the beginning of a period is ignored. With regard to this aspect a modified CLSP with linked lot-sizes of adjacent periods, denoted by CLSPLA, can mathematically be stated as follows:

$$\min \sum_{j=1}^J \sum_{t=1}^T [s_j(x_{jt} - z_{jt}) + h_j I_{jt}] \quad (7)$$

subject to

$$I_{j,t-1} + q_{jt} - I_{jt} = d_{jt} \quad (j=1,\dots,J; t=1,\dots,T) \quad (8)$$

$$\sum_{j=1}^J p_j q_{jt} \leq C_t \quad (t=1,\dots,T) \quad (9)$$

$$C_t x_{jt} - p_j q_{jt} \geq 0 \quad (j=1,\dots,J; t=1,\dots,T) \quad (10)$$

$$\sum_{j=1}^J z_{jt} \leq 1 \quad (t=1,\dots,T) \quad (11)$$

$$z_{jt} - x_{j,t-1} \leq 0 \quad (j=1,\dots,J; t=1,\dots,T) \quad (12)$$

$$z_{jt} - x_{jt} \leq 0 \quad (j=1,\dots,J; t=1,\dots,T) \quad (13)$$

$$z_{j,t-1} + z_{jt} \leq 1 \quad (j=1,\dots,J; t=1,\dots,T) \quad (14)$$

$$I_{jt}, q_{jt} \geq 0 \quad (j=1,\dots,J; t=1,\dots,T) \quad (15)$$

$$x_{jt}, z_{jt} \in \{0, 1\} \quad (j=1,\dots,J; t=1,\dots,T) \quad (16)$$

where

x_{jt} = a binary variable indicating whether a quantity is produced for item j in period t ($x_{jt}=1$) or not ($x_{jt}=0$),

z_{jt} = a binary variable indicating whether the quantities of item j in period $t-1$ and period t are linked ($z_{jt}=1$) or not ($z_{jt}=0$),

W.l.o.g. it is $I_{j0}=0$, $x_{j0}=0$, $z_{j0}=z_{j1}=0$ for all $j=1,\dots,J$.

The objective function (7) computes only setup costs for an item j in a period t if no linking with $q_{j,t-1}$ (i.e. the production quantity of the preceeding period of the item) is performed. (8) corresponds to ordinary inventory balance constraints. (9) secures feasibility with respect to the machine capacity. (10) couples the production decisions with the setup state of the machine. (11) to (13) secure that only one product can be produced at the end of a period and produced further (linking) in the following period. (14) avoid that the same quantity will be considered as produced at the beginning *and* at the end of a period t ; i.e. the quantity can either be linked with a quantity of the preceeding period $t-1$ or with a quantity of the succeeding period $t+1$.

Note, the CLSPLA is equivalent to the CLSP if we set $z_{jt}=0$ for $j=1,\dots,J$, and $t=1,\dots,T$.

As in the CLSP a large time scale is assumed, thus we expect that more than one item will be (usually) scheduled in a period, i.e. the required capacity of a lot will be strictly less than the available period capacity. Let us consider the (unusual) case where in the period t *only* the item j is scheduled, and the item j is also scheduled in the period $t-1$, and $t+1$. Thus, the setup state may be preserved for item j from the end of period $t-1$ up to the beginning of period $t+1$. To preserve the setup state two links are necessary for item j which is not allowed due to (14). However, linked production quantities may be produced as one splitted or non-splitted lot.

If a CLSP is solved, then the provided schedule can be modified by performing the links afterwards, which reduces setup costs. Moreover, holding costs are saved, if a quantity of a linked lot can be "right-shifted". We consider Example 2.

Example 2: Let $J=4$, $T=4$, $C_t=100$ for $t=1$ to 4, and the other data as provided in Table 3.

Table 3

t	1	2	3	4	p_j	h_j	s_j
d_{1t}	20	10	30	20	1	1	200
d_{2t}	30	10	30	30	1	1	150
d_{3t}	0	30	10	60	1	1	100
d_{4t}	20	20	0	10	1	1	150

We solve this problem as a CLSP instance as well as a CLSPLA to optimality. The CLSP solution can be modified by linking and "right-shifting". In Table 4 the optimal CLSP and CLSPLA solution, and the modified CLSP solution are entered. For example, a right-shifting of "30 units" occurs for $q_{23}=60$.

Table 4 CLSP, modified CLSP, and CLSPLA solution

	t	1	2	3	4	Z^*
CLSP	q_{1t}	40	0	40	0	1320
	q_{2t}	40	0	60	0	
	q_{3t}	0	40	0	60	
	q_{4t}	10	30	0	0	
modified CLSP	q_{1t}	40	0	40	0	1130
	q_{2t}	40	0	30 \cup	30	
	q_{3t}	0	30 \cup	10	60	
	q_{4t}	10 \cup	30	0	0	
CLSPLA	q_{1t}	20 \cup	60	0	0	1000
	q_{2t}	30	10 \cup	60	0	
	q_{3t}	0	30	10 \cup	60	
	q_{4t}	50	0	0	0	

The modified CLSP solution is 11.3% more costly than the CLSPLA solution. Thus the solution quality can be improved substantially by integrating (semi-) sequencing.

Note that the sets of feasible solutions of the models are identical whereas a (modified) optimal solution of the CLSP can be more costly, so the following inequalities can be stated:

$$Z_{\text{CLSP}}^* \geq Z_{\text{modified CLSP}}^* \geq Z_{\text{CLSPLA}}^*$$

Since the CLSPLA contains more binary variables than the CLSP it may be more difficult to determine an optimal solution in reasonable time. However, in practical applications fast heuristics have to be applied. So a CLSPLA approach which computes in a reasonable time a solution may be more attractive than a CLSP based exact or heuristic method.

Note, recently a similar model with machine state preserving has been presented in [9] which is a special case of the multi-machine case with setup times introduced by [4].

3. A Backward-Oriented Heuristic for the CLSPLA

In the following we describe a simple heuristic for the CLSPLA which starts with scheduling at the planning horizon and step backwards to the first period. The lot-size decisions are performed by a simple priority rule which consists of a convex combination of holding and setup costs.

To define the priority rule we have to introduce some additional notations.

The cumulative demand of item j from period t to the horizon T which has to be satisfied in the periods $t, \dots, 1$ is defined by

$$D_{jt} := \max \left\{ 0, \sum_{\tau=t}^T (d_{j\tau} - q_{j\tau}) \right\} \quad \text{for } j=1, \dots, J \text{ and } t=1, \dots, T.$$

The total still required capacity is specified by

$$\text{TRC} := \sum_{j=1}^J p_j D_{j1}.$$

The available capacity in period t will be computed as follows:

$$AC_t = C_t - \sum_{j=1}^J p_j q_{jt} \quad \text{for } t=1, \dots, T$$

The cumulative capacity from period $\tau=1$ to period $\tau=t$ is denoted by

$$CC_t := \sum_{\tau=1}^t C_\tau \quad \text{for } t=1, \dots, T.$$

Scheduling will be done backward oriented, i.e. first the lot-sizes are determined in period $t=T$, then in period $t=T-1$, and so on. The decisions depend on a priority rule.

To derive a priority value which is based on "savings" we distinguish four cases:

a) There is unsatisfied demand of item j in period t and the available capacity in period t is greater or equal than the capacity which will be required if item j is scheduled in period t , i.e. $AC_t \geq p_j D_{jt} > 0$. Thus, to schedule item j in period t and not in period $t-1$ saves holding costs $h_j D_{jt}$ and incurs setup costs s_j .

b) It is $AC_t < p_j D_{jt} > 0$. If a setup occurs for item j in period t item j will be scheduled in period t and $t-1$ with linking, i.e. $z_{jt} = 1$. Thus, setup costs s_j are incurred in period $t-1$ and holding costs $h_j D_{j,t-1}$ ($=h_j D_{jt} + d_{j,t-1}$) will be saved because the amounts AC_t/p_j and $D_{j,t-1} - AC_t/p_j$ are not scheduled before t or $t-1$, respectively.

c) Item j is already scheduled in period t , there exists a positive demand of item j in period $t-1$, and there is no linking performed for item j in period $t+1$, i.e. $z_{j,t+1} = 0$, $q_{jt} > 0$, and $d_{j,t-1} > 0$. Thus a link in period t for item j avoids setup costs s_j and holding costs $h_j d_{j,t-1}$ because the amount is not scheduled in period $t-2$. After linking is performed between period t and period $t-1$ no more changes of the schedule from period t up to period T will be performed. Thus if we leave period t and it is $AC_t > 0$ we have to perform a feasibility check, that is, the total still required capacity must be less or equal the available capacity from period 1 up to period $t-1$, i.e. $TRC \leq CC_{t-1}$

d) Linking does not improve the solution quality (i.e. $D_{j,t} = D_{j,t-1} = 0$) or linking leads to infeasibility.

Thus, we define the following priority value for item j in period t

$$r_{jt} := \begin{cases} (1-\gamma)h_j D_{jt} - \gamma \cdot s_j & \text{if } (AC_t < p_j D_{jt} > 0) \wedge (x_{jt} = 0) \\ (1-\gamma)h_j D_{j,t-1} - \gamma \cdot s_j & \text{if } (AC_t \geq p_j D_{jt} > 0) \wedge (x_{jt} = 0) \\ (1-\gamma)h_j d_{j,t-1} + \gamma \cdot s_j & \text{if } (z_{j,t+1} = 0) \wedge (x_{jt} = 1) \wedge (d_{j,t-1} > 0) \wedge (TRC \leq CC_{t-1}) \\ -\infty & \text{otherwise} \end{cases}$$

where $0 \leq \gamma \leq 1$. The larger the r_{jt} the more preferable it is to schedule the item j in period t . Thus the item with the largest priority value will be scheduled (**priority rule**).

By the parameter γ we will control the expected lot-sizes, e.g. if $\gamma=1$ we expect large lot-sizes for items with high setup costs.

A formal description of the backward oriented method, denoted by BACLSPLA, is given in the following:

BACLSPLA

INITIALIZATION:

for $j=1, \dots, J$, $t=1, \dots, T$ **do begin** $x_{jt}:=0$; $q_{jt}:=0$; $D_{jt} := \sum_{\tau=t}^T d_{j\tau}$; $z_{jt}:=0$ **end**

for $j=1, \dots, J$ **do** $z_{j,T+1}:=0$;

$TRC := \sum_{j=1}^J p_j D_{j1}$;

for $t=1, \dots, T$ **do** $CC_t := \sum_{\tau=1}^t C_{\tau}$;

SCHEDULING:

$t:=T$;

$AC_t := C_t$

while $t > 2$ **do**

begin

for $j=1, \dots, J$ **do** compute r_{jt} ;

i = item with maximal priority value;

if $x_{it}=1$ **then begin** $z_{it}=1$; $t:=t-1$ **end**

else

begin if $p_i D_{it} > AC_t$ **then**

begin $q_{it}:=AC_t/p_i$; $TRC:=TRC - p_i q_{it}$; $x_{it}:=1$; $z_{it}:=1$; $t:=t-1$; $AC_t:=C_t$;

for $\tau=1, \dots, t$ **do** $D_{i\tau}:=D_{i\tau} - q_{it}$

end;

end;

$q_{it}:=\min\{C_t/p_i, D_{it}\}$; $AC_t:=AC_t - p_i q_{it}$; $TRC:=TRC - p_i q_{it}$; $x_{it}:=1$;

for $\tau=1, \dots, t$ **do** $D_{i\tau}:=D_{i\tau} - q_{it}$;

end;

for $j=1, \dots, J$ **do if** $D_{j1} > 0$ **then begin** $x_{j1}:=1$; $q_{j1}:=D_{j1}$ **end;**

If $\sum_{j=1}^J p_j q_{j1} > C_1$ **then** $Z := \infty$

else evaluate solution.

Note, if for an instance a feasible solution exists, i.e. $\sum_{\tau=1}^t \sum_{j=1}^J p_j d_{j\tau} \leq CC_t$ for $t=1, \dots, T$,

BACLSPLA, computes a feasible solution due to the feasibility check.

To see how BACLSPLA works we start to solve the instance in Example 2 with $\gamma = 0.5$.

Before starting the scheduling phase it is $TRC = 330$, $t = 4$ and D_{jt} as given in Table 5.

Table 5

t	1	2	3	4
D_{1t}	80	60	50	20
D_{2t}	100	70	60	30
D_{3t}	100	100	70	60
D_{4t}	50	30	10	10

The following priority values for the items $j=1, \dots, 4$ will be computed in the first iteration of the while-loop:

$$r_{14} = (1 - 0.5) \cdot 1 \cdot 20 - 0.5 \cdot 200 = -90$$

$$r_{24} = (1 - 0.5) \cdot 1 \cdot 30 - 0.5 \cdot 150 = -60$$

$$r_{34} = (1 - 0.5) \cdot 1 \cdot 60 - 0.5 \cdot 100 = -20$$

$$r_{44} = (1 - 0.5) \cdot 1 \cdot 10 - 0.5 \cdot 150 = -70$$

Due to the priority rule item $j=3$ will be scheduled in period $t=4$, i.e. $x_{34}=1$, and $q_{34}=60$. In the beginning of the second iteration of the while-loop it is $TRC = 330 - 60 = 270$ which is less than $CC_3 = 300$ (feasibility check), thus the new priority value of item $j=3$ in period 4 is $r_{34} = 0.5 \cdot 100 + (1 - 0.5) \cdot 10 = 55$. Thus, item $j=3$ will be selected a second time, i.e. $t=3$, $z_{33}=1$, $x_{33}=1$, $q_{33}=10$, and so on.

If we proceed further a non-optimal schedule which is 1150 costly will be computed. Trying to get a better solution, we have also used $\gamma=0$, $\gamma=25$, ..., and $\gamma=1$. The corresponding objective function values are reported in Table 6.

Table 6

γ	0.0	0.25	0.50	0.75	1.0
Z	1250	1000	1150	1150	1150

For our instance, the optimal solution will be computed if we take $\gamma=0.25$. However, the solution quality depends heavily on the choice of the parameter value. Therefore, we apply a search procedure for the parameter value $\gamma \in [\underline{\gamma}, \bar{\gamma}] \subseteq [0,1]$. We start with $\gamma=\underline{\gamma}$. Then γ will be increased about $\Delta:=(\bar{\gamma}-\underline{\gamma})/b$, where b is an integer greater than 2, as long as an improvement of the objective function value will be computed by BACLSPLA or $\gamma=\bar{\gamma}$. If an improvement has been achieved a more detailed search will be started with $\Delta:=\Delta/b$, $\underline{\gamma}:=\max\{\Delta, \delta-\Delta(b-1)\}$, and $\bar{\gamma}=\min\{1-\Delta, \delta+\Delta(b-1)\}$; otherwise the search procedure stops.

Now, let $b=4$, $\underline{\gamma}=0$, $\bar{\gamma}=1$, thus $\Delta:=(\bar{\gamma}-\underline{\gamma})/4=0.25$. Furthermore, let δ be the parameter value where BACLSPLA has been computed the best solution. For our instance, γ will be increased up to 0.5, and δ will be 0.25. Then a new (and last) search will be started with $\Delta:=0.05125$, $\underline{\gamma}=0.05125$, and $\bar{\gamma}=0.44875$.

A formal description of the search method is given in the following:

Let $\gamma:=0$, $\bar{\gamma}:=1$, $\delta:=-1$, $\Delta:=0.25$, $b:=4$, $\bar{Z}:=\infty$, and improvement=false;

while $\gamma \leq \bar{\gamma}$ **do**

begin

if $\gamma \neq \delta$ **then** $Z:=$ objective function value of BACLSPLA solution

if $Z < \bar{Z}$ **then**

begin $\bar{Z}:=Z$; improvement:=true; $\delta:=\gamma$ **end**

else if improvement **then**

begin $\Delta:=\Delta/b$; $\underline{\gamma}:=\max\{\Delta, \delta-\Delta(b-1)\}$; $\bar{\gamma}:=\min\{1-\Delta, \delta+\Delta(b-1)\}$; $\gamma:=\underline{\gamma}$ improvement:=false

end;

$\gamma:=\gamma+\Delta$

end;

Note, the parameter values $\gamma = \Delta \cdot b$, $\gamma = -\Delta \cdot b$, and $\gamma = \delta$ are used once, only

It should be noted here that backward lot-sizing and scheduling methods in the capacitated case are from a conceptual point of view superior to forward-oriented ones like e.g. the Dixon, Silver heuristic: There is no need for complicated and time-consuming look ahead procedures in order to secure resource feasibility.

4. Computational Study

The computational performance of BACLSPLA is compared with the Dixon-Silver heuristic. In [2] 120 instances are reported. There are three sets of instances which differ in the number of items and periods, (J,T), as (50,8), (20,20), and (8,50), respectively.

The instances of a set differ regarding three factors, i.e. capacity utilization (U), capacity requirements (production speeds) (C), and demand (supply) variation (S).

For each factor there are two levels. These levels are low (L) and high (H) for

- capacity utilization (LU, HU) and
 - demand variation (LS,HS);
- constant (C) and varying (V) for
- capacity requirements (CC,VC).

The instances are solved with the Dixon-Silver heuristic, denoted with DS, and with BACLSPLA. For a given DS solution setups are *reduced* a posteriori, by combining two lots of adjacent periods (for the item with the highest setup costs, if there is a choice). The corresponding objective function value is denoted by \bar{Z}_{DS} . The average deviation (of 5 instances per entry) between the *reduced costs* of DS and the costs of BACLSPLA (i.e. $100 \cdot (\bar{Z}_{DS} - \bar{Z}_{BACLSPLA}) / \bar{Z}_{BACLSPLA}$) are reported in Table 7.

Table 7 Comparison of the solution quality of DS and BACLSPLA

(J,T)	(50,8)	(20,20)	(8,50)
LU-CC-LS	3.29	6.13	9.07
LU-CC-HS	1.30	1.69	4.40
LU-VC-LS	4.55	6.07	10.87
LU-VC-HS	1.69	-0.48	6.05
HU-CC-LS	0.08	15.65	12.52
HU-CC-HS	-0.89	5.33	10.92
HU-VC-LS	-2.14	14.90	19.62
HU-VC-HS	-2.66	2.49	8.89
HU average	-1.40	9.59	12.99
LU average	2.71	3.35	7.60
LS average	1.45	8.11	13.02
HS average	-0.14	2.26	7.56
CC average	0.95	7.20	9.23
VC average	0.36	5.74	11.36
Total average	0.65	6.47	10.29

The solution quality of DS and BACLSPLA is for a large number of items very nearly the same. This may be reasoned because for a large number of items it is "easy" to determine afterwards a "good" linking between production quantities of adjacent periods. Thus the solution quality is not inferior if a CLSP-solution is performed in a first step and then in a second a step sequencing is done. For a medium (~ 20) and a small number (~ 8) of items the solution quality of BACLSPLA is substantial better than the solution quality of DS, i.e. the integrated sequencing is very important regarding the solution quality. For example, for the instances with high capacity utilization, varying capacity requirements, and low demand variation, denoted with HU-VC-LS, the DS solutions are very poor, i.e. on the average they are 19.62 % more costly than the corresponding CLSPLA solutions.

BACLSPLA has been coded in Turbo Pascal 6.0 from Borland. Table 8 gives the average computation times (of 40 instances per entry) on a 486 machine with 50Mhz.

Table 8 Computation times of BACLSPLA in seconds

(J,T)	(50,8)	(20,20)	(8,50)
average	4.5	1.2	0.73

Note that the number of items has a significant effect on the computation time. Nevertheless, the method is very fast.

For BACLSPLA the average of the total number of execution as well as the average number of executions until the best solution has been computed are entered in Table 9.

Table 9 Average number of BACLSPLA executions

(J,T)	(50,8)	(20,20)	(8,50)
total	14.48	13.85	14.9
best solution	7.48	6.93	7.90

Thus, on the average the first (best) solution is computed by BACLSPLA for the instances with 50 items and 8 periods in $4.5/14.48=0.31$ ($0.31 \cdot 7.48=2.32$) seconds.

5. Summary

To integrate sequencing in lot-sizing can be very attractive. A mixed-integer formulation is presented where lot-sizes of adjacent periods can be linked (semi-scheduling), denoted by CLSPLA. The CLSPLA can be solved efficiently by a backward oriented approach where lot-sizing and linking depends on a priority rule. Especially for instances with a small to medium

size number of items the method solves the CLSPLA more efficient than the Dixon-Silver heuristic.

Acknowledgement

The author is grateful to Andreas Drexel for his helpful support and to Jörg Latteier for his assistance in the computational study.

References

- [1] **Barany, I., van Roy, T.J., and Wolsey, L.A.**, "Strong formulations for multi-item capacitated lot-sizing", *Management Science*, Vol. 30 (1984), pp. 1255-1261.
- [2] **Cattrysse, D., Maes, J., and Van Wassenhove, L.N.**, "Set partitioning and column generation heuristics for capacitated lotsizing", *European Journal of Operational Research*, Vol. 46 (1990), pp. 38-47.
- [3] **Diaby, M., Bahl, H.C., Karwan, M.H., and Zionts, S.**, "A Lagrangean relaxation approach for very-large-scale capacitated lot-sizing", *Management Science*, Vol. 38 (1992), pp. 1329-1340.
- [4] **Dillenberger, C., Escudero, L.F., Wollensak, A., and Zahng, W.**, "On solving a large-scale resource allocation problem in production planning", in: *Operation research in production planning and control*, Fandel, G. et al. (eds.), Springer (Berlin) 1992, pp. 105-119.
- [5] **Dixon, P. S., and Silver, E.A.**, "A heuristic solution procedure for multi-item single-level, limited capacity, lot-sizing problem", *Journal of Operations Management* 2/1 (1981), pp. 23-39.
- [6] **Dogramaci, A., Panayiotopoulos, J.C., and Adam, N.R.**, "The dynamic lot-sizing problem for multiple items under limited capacity", *AIIE Transactions*, Vol. 13 (1981), pp. 294-303.

- [7] **Gelders, L.F., Maes, J., and Van Wassenhove, L.N.**, "A branch and bound algorithm for the multi-item single level capacitated dynamic lot-sizing problem", in: Multi-stage production planning and inventory control, Axsäter, S. et al. (eds.), Springer (Berlin) 1986, pp. 92-108.
- [8] **Günther, H.O.**, "Planning lot sizes and capacity requirements in a single stage production system", European Journal of Operational Research, Vol. 31 (1987), pp. 223-231.
- [9] **Haase, K.**, "Lotsizing and Scheduling for Production Planning", Springer (Berlin) 1994.
- [10] **Lambrecht, M.R., and Vanderveken, H.**, "Heuristic procedures for the single operations, multi-item loading problem", AIIE Transactions, Vol. 11 (1979), pp. 319-326.
- [11] **Thizy, J.M., and Van Wassenhove, L.N.**, "Lagrangian relaxation for the multi-item capacitated lot-sizing problem: A heuristic implementation", IIE Transactions, Vol 17 (1985), pp. 308-313.